RVS COLLEGE OF ARTS AND SCIENCE(AUTONOMOUS)
SULUR-COIMBATORE-641402

Reg. No	
Name	
Class	**III-B.COM(IT)**
Subject	**PHP PROGRAMMING**

B. Com (IT) Degree

Examinations –OCTOBER 2025

REGISTER NO. _____

Certified that it is a Bonafide record of PHP PROGRAMMING Practical Report done and submitted by _____ of _____ for the Practical Examination / Viva – Voce held at RVS College of Arts and Science (Autonomous), Sulur, Coimbatore.

_____ _____

Staff – in – charge **Director/ HoD**

Submitted for the practical Examination / Viva – Voce held on _____.

---------- ----------

Internal Examiner **External Examiner**

CONTENTS

S.NO.	PARTICULARS	DATE	PAGE.NO	SIGN
1	Product Discount Application			
2	Future Value Application			
3	Enhance the product manager application			
4	Enhance the guitar shop application			
5	Fetching category and guitar shop database in phpmyadmin			
6	Data manipulation on the product table			
7	Future value calculation using a function with parameters			
8	GET_Array Method Using PHP			
9	POST_Array Method Using PHP			
10	Relational and logical operators in php			
11	Strings in php			
12	Numbers in php			
13	Build in functions in php			
14	While loop in php			
15	Arrays in php			

1.Product Discount Application

AIM:

Built the Product Discount Application

ALGORITHM:

STEP 1 : Turn On The PC.

STEP 2 : Open The **XAMPP**,And Start **APACHE**&**MYSQL**.

STEP 3 : Open **Notepad** to code the Function.

STEP 4 : Start the code with **<HTML>**.

STEP 5 : Create the form actionand finish the HTML code with save as file.HTML extension.

STEP 6 : Open the PHP tag <?php and create php code

STEP 7 :Declare the variable $product_description,$list_price,$discount_percent

STEP 8 And create the empeddedphp code for display process

STEP 9 : Finish The Coding Function.

STEP 10 : To Run a Program Open **Chrome** and Type **localhost/Folder Name**. And Select The File To View An Output In Parent Directory.

STEP 11 : Stop The **APACHE**&**MYSQL** And Close The **XAMPP**.

1. Product Discount Application(HTML FILE)

```
<html >
<head>
<title>Product Discount Calculator</title>
</head>
<body>
<div id="content">
<h1>Product Discount Calculator</h1>
<form action="p1.php" method="post">
<div id="data">
<label>Product Description:</label>
<input type="text" name="product_description"/>
<br />
<label>List Price:</label>
<input type="text" name="list_price"/><br />
<label>Discount Percent:</label>
<input type="text" name="discount_percent"/>%<br />
</div>
<div id="buttons">
<label> </label>
<input type="submit" value="Calculate Discount" />
<br />
</div>
</form>
</div>
</body>
</html>
```

1. Product Discount Application(PHP FILE)

```
<?php
$product_description = $_POST['product_description'];
$list_price = $_POST['list_price'];
$discount_percent = $_POST['discount_percent'];
$discount = $list_price * $discount_percent * .01;
$discount_price = $list_price - $discount;
$list_price_formatted =
"$".number_format($list_price, 2);
$discount_percent_formatted = $discount_percent."%";
$discount_formatted = "$".number_format($discount, 2);
$discount_price_formatted =
"$".number_format($discount_price, 2);
?>
<html >
<head>
<title>Product Discount Calculator</title>
<link rel="stylesheet" type="text/css" href="main.css"/>
</head>
<body>
<div id="content">
<h1>Product Discount Calculator</h1>
<label>Product Description:</label>
<span><?php echo $product_description; ?>
</span><br />
<label>List Price:</label>
<span><?php echo $list_price_formatted; ?>
</span><br />
<label>Standard Discount:</label>
<span><?php echo $discount_percent_formatted; ?>
</span><br />
<label>Discount Amount:</label>
<span><?php echo $discount_formatted; ?>
</span><br />
<label>Discount Price:</label>
<span><?php echo $discount_price_formatted; ?>
</span><br />
</div></body></html>
```

OUTPUT:

Product Discount Calculator

Product Description: guitar

List Price: 2500

Discount Percent: 10 **%**

Calculate Discount

Product Discount Calculator
Product Description: guitar
List Price: $2,500.00
Standard Discount: 10%
Discount Amount: $250.00
Discount Price: $2,250.00

RESULT:
The Program Will Be Executed, The Output Is Displayed.

2. Future Value Calculation

AIM:

Enhance the Future value Application to display the result on same page

ALGORITHM:

Step 1 : Turn On The PC.

Step 2 : Open The **XAMPP**,And Start **APACHE**&**MYSQL**.

Step 3 : Open **Notepad**

Step 4 : Start the code with **<HTML>**.

Step 5 : Create the form action and finish the HTML code with save as file.HTML extension.

Step 6 : Open the PHP tag <?php and create php code

Step 7 : Declare the variable $investment amount,$interest rate,$future value

Step 8 : And create the empeddedphp code for display process

Step 9 : Finish The Coding Function.

Step 10 : To Run a Program Open **Chrome** and Type **localhost/Folder Name**. And Select The File To View An Output In Parent Directory.

Step 11 : Stop The **APACHE**&**MYSQL** And Close The **XAMPP**.

2. Futurevalue Calculation(PHP FILE)

```
<html>
<head>
<title>Future Value Calculator</title>
</head>
<body>
<h2>Future Value Calculator</h2>
<form method="post">
      Present Value (PV): <input type="number" step="0.01" name="pv" required><br><br>
      Annual Interest Rate (%): <input type="number" step="0.01" name="rate"
required><br><br>
      Number of Periods (Years): <input type="number" name="n" required><br><br>
<input type="submit" value="Calculate Future Value">
</form>

<?php
   if ($_SERVER["REQUEST_METHOD"] == "POST")
 {
      $pv = $_POST['pv'];
      $rate = $_POST['rate'];
      $n = $_POST['n'];

      $r = $rate / 100;
      $fv = $pv * pow((1 + $r), $n);

      echo "<h3>Result:</h3>";
      echo "Present Value (PV): $pv<br>";
      echo "Interest Rate: $rate% <br>";
      echo "Periods (Years): $n <br>";
      echo "<strong>Future Value (FV): " .number_format($fv, 2) . "</strong>";
   }
   ?>
</body>
</html>
```

Output:
Present Value (PV): 50000

Annual Interest Rate (%): 10

Number of Periods (Years): 50

Result:
Present Value (PV): 50000
Interest Rate: 10%
Periods (Years): 50
Future Value (FV): 5,869,542.64

RESULT:
The Program Will Be Executed, The Output Is Displayed.

3. Enhance The Product Manager Application

AIM:

Enhance the Product Manager Application

ALGORITHAM:

Step 1: Set Up Your XAMPP Environment

1. **Install XAMPP**: If you haven't already, download and install XAMPP from the official site.
2. **Start XAMPP**: Launch the XAMPP Control Panel and start the **Apache** and **MySQL** services.

Step 2: Create the Database and Table

1. **Open phpMyAdmin**: In your browser, go to http://localhost/phpmyadmin.
2. **Create Database**:
   - Click on the **Databases** tab.
   - Enter product_manager as the database name and click **Create**.
3. **Create Table**: Use the following SQL to create a products table:

Import SQL File (Optional): If you have a SQL file, you can import it by:

- Clicking on the **product_manager** database.
- Going to the **Import** tab.
- Selecting the SQL file and clicking **Go**.

Step 3: Create the PHP Application

1. **Set Up Project Directory**:
   - Navigate to the htdocs folder in your XAMPP installation (usually found in C:\xampp\htdocs).
   - Create a new folder named book_apps.
2. **Create PHP Files**: Create the following files in the book_apps directory.

Step 4: Test the Application

1. **Open the Application**: In your browser, go to http://localhost/book_apps/ch4_product_manager/index.php.
2. **View Products**: You should see a list of products (if you have any in the database).

Step 5: Add Functionality (Optional)

You can extend the application by adding forms for adding, editing, and deleting products. Here's a simple way to add a form to insert products:

3. Enhance The Product Manager Application

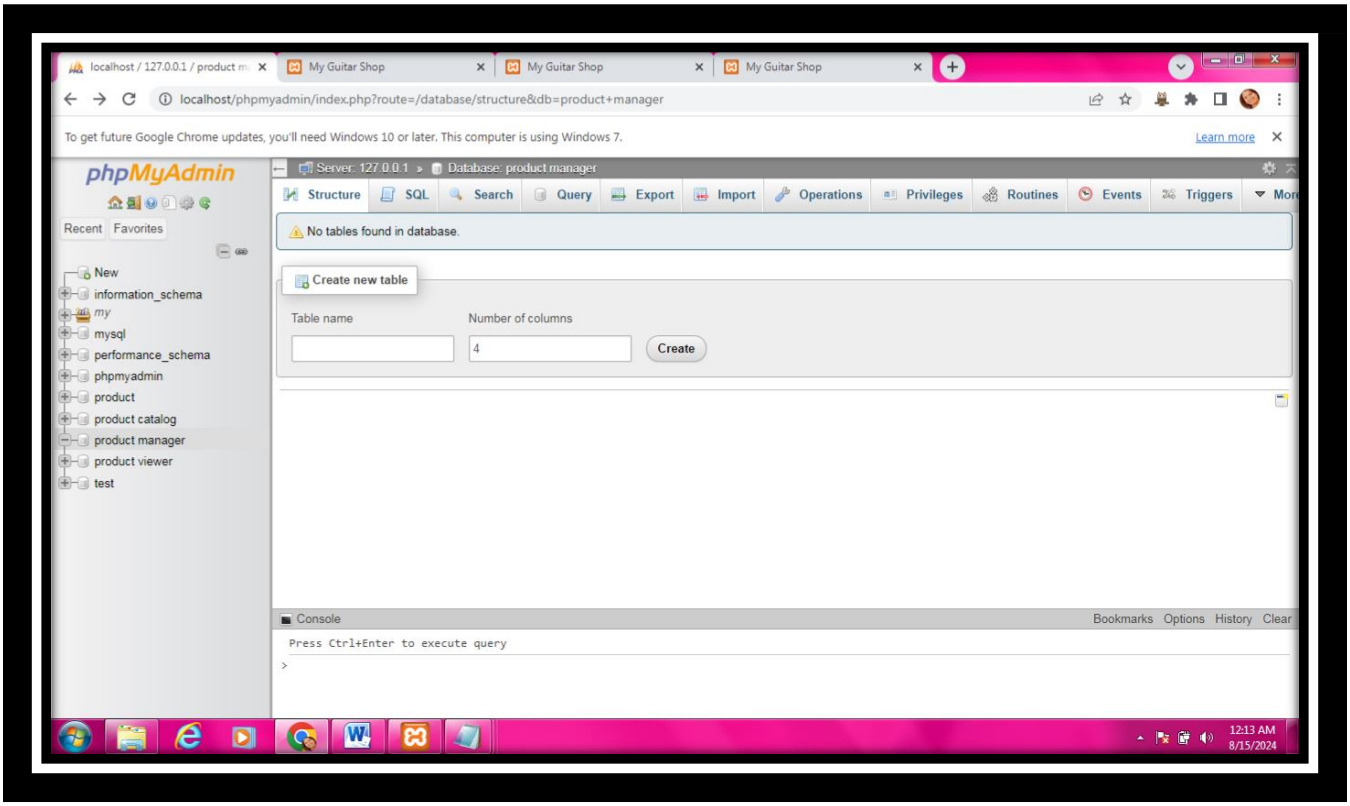
```
-- Create the database
    CREATE DATABASE product_manager;

-- Select the database
    USE product_manager;

-- Create categories table
    CREATE TABLE categories (categoryID INT AUTO_INCREMENT PRIMARY KEY,
    categoryName VARCHAR(255) NOT NULL

-- Create products table
     CREATE TABLE products (
          productID INT AUTO_INCREMENT PRIMARY KEY,
          categoryID INT NOT NULL,
          productCode VARCHAR(10) NOT NULL,
          productName VARCHAR(255) NOT NULL,
          listPrice DECIMAL(10,2) NOT NULL,
          FOREIGN KEY (categoryID) REFERENCES categories(categoryID)
          ON DELETE CASCADE
           ON UPDATE CASCADE
);
```

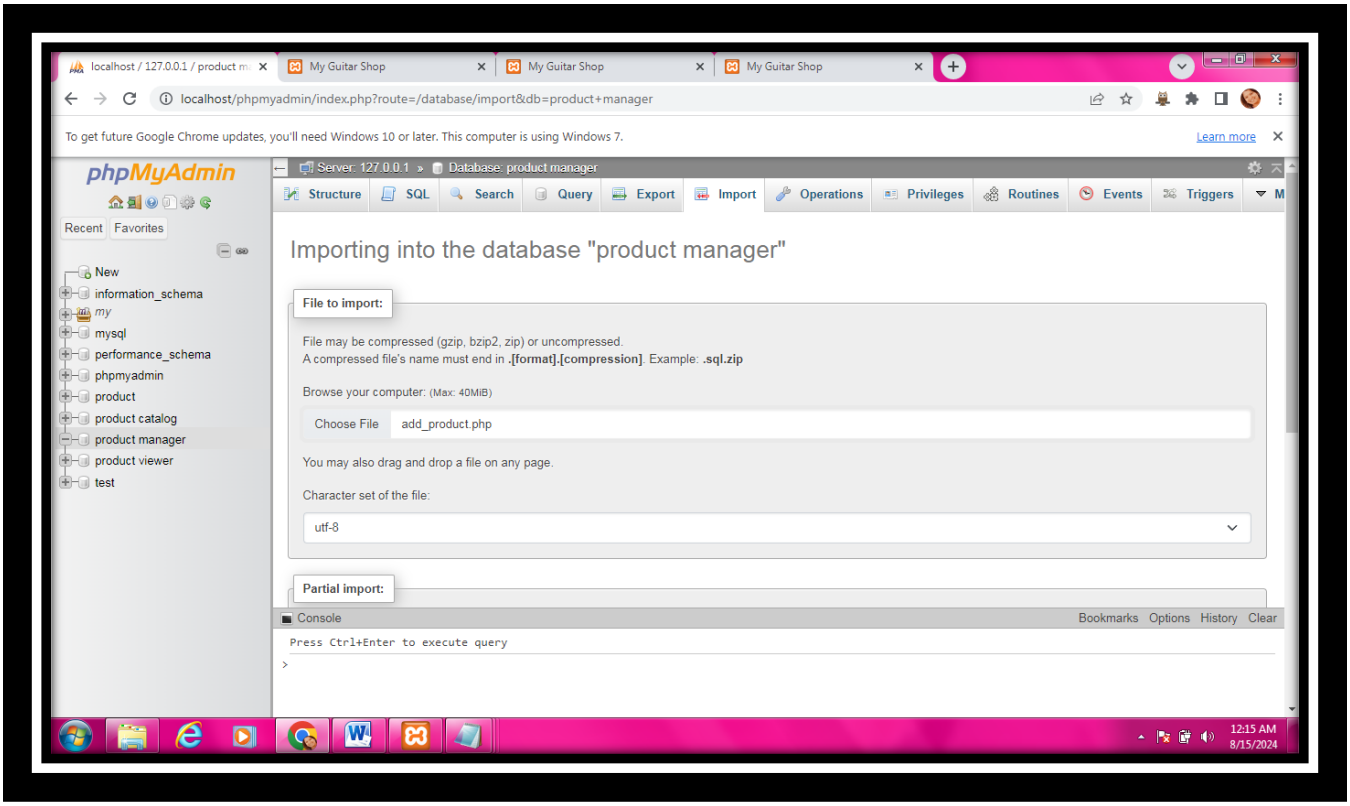
Database connectivity program of php

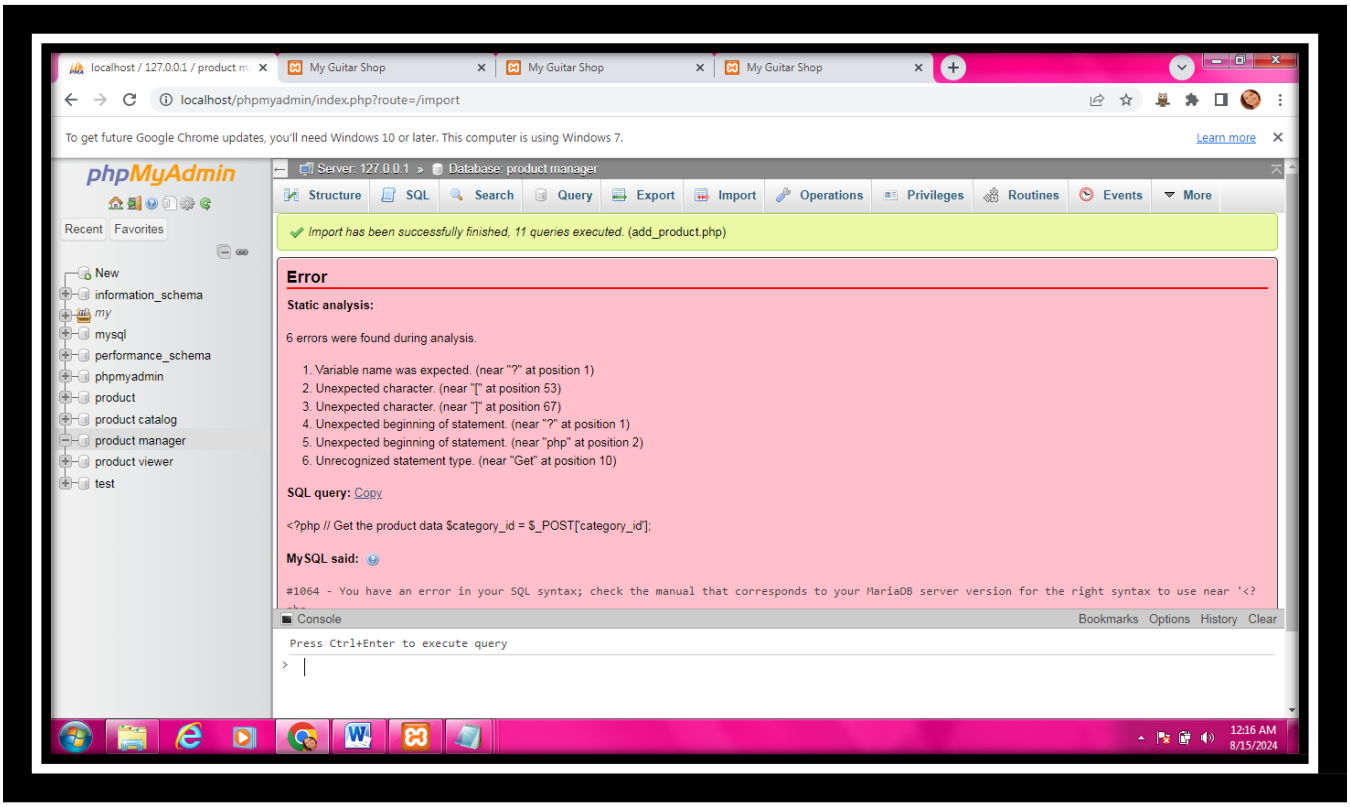
```
<?php
   $dsn = 'mysql:host=localhost;dbname=product_manager';
   $username = 'root';
   $password = '';
   try {
```

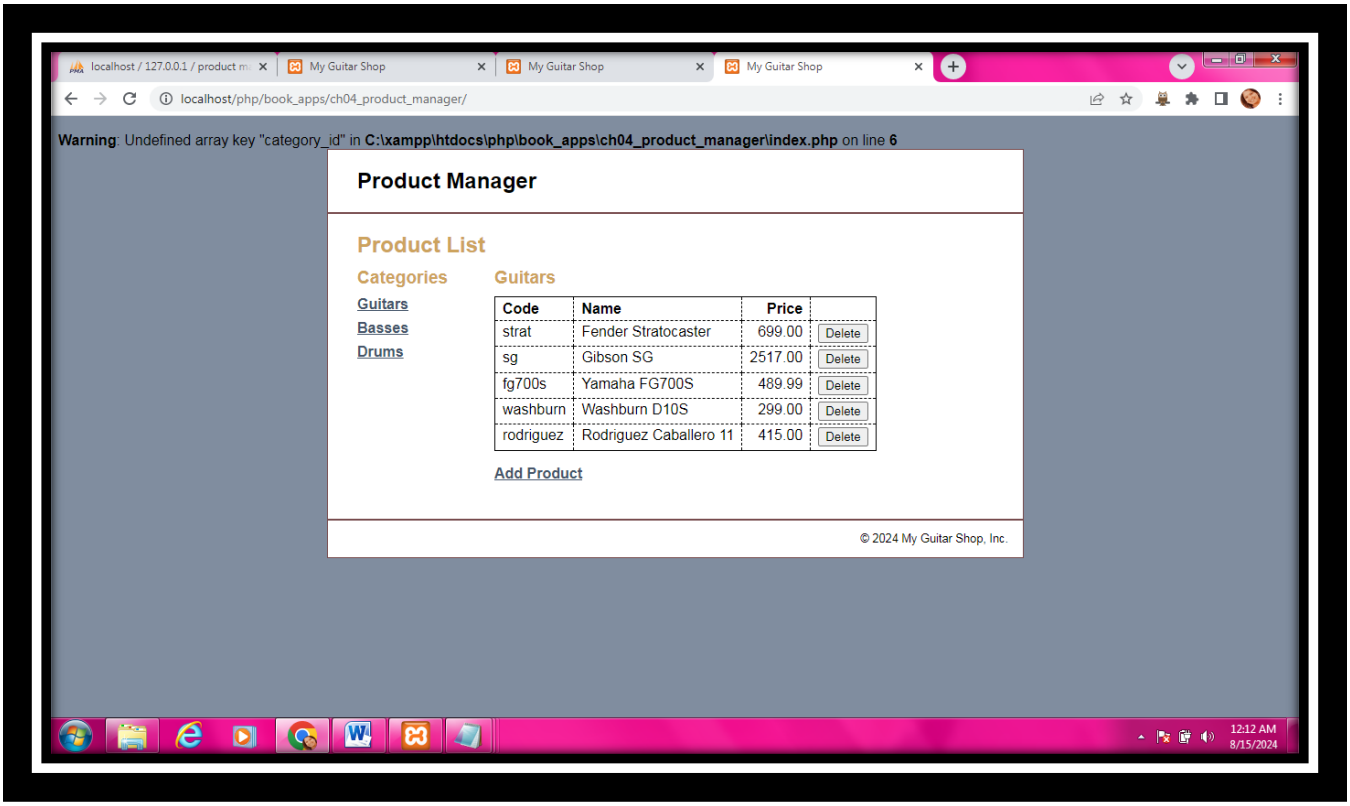
```
        $db = new PDO($dsn, $username, $password);
    }
catch (PDOException $e) {
        $error_message = $e->getMessage();
        include('database_error.php');
        exit();
    }
?>
```

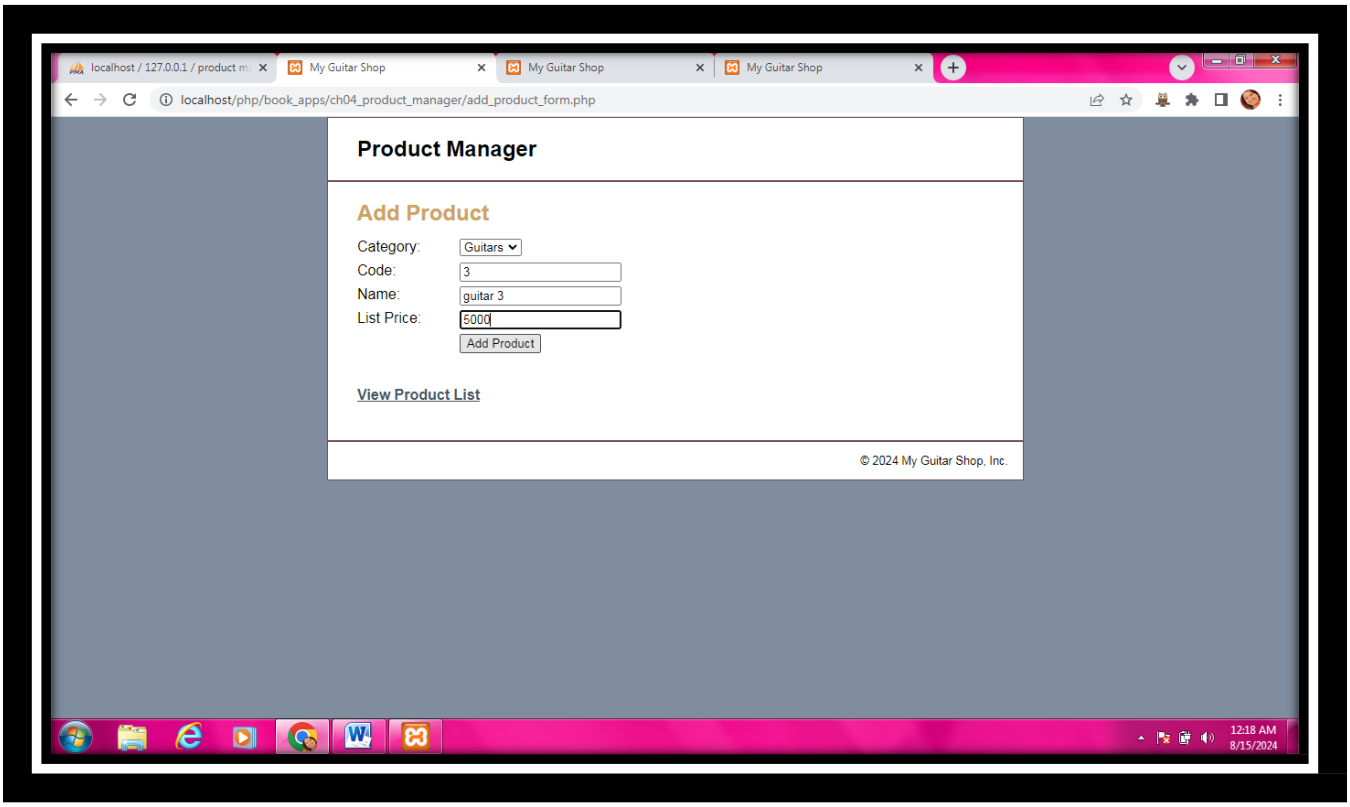
OUTPUT:

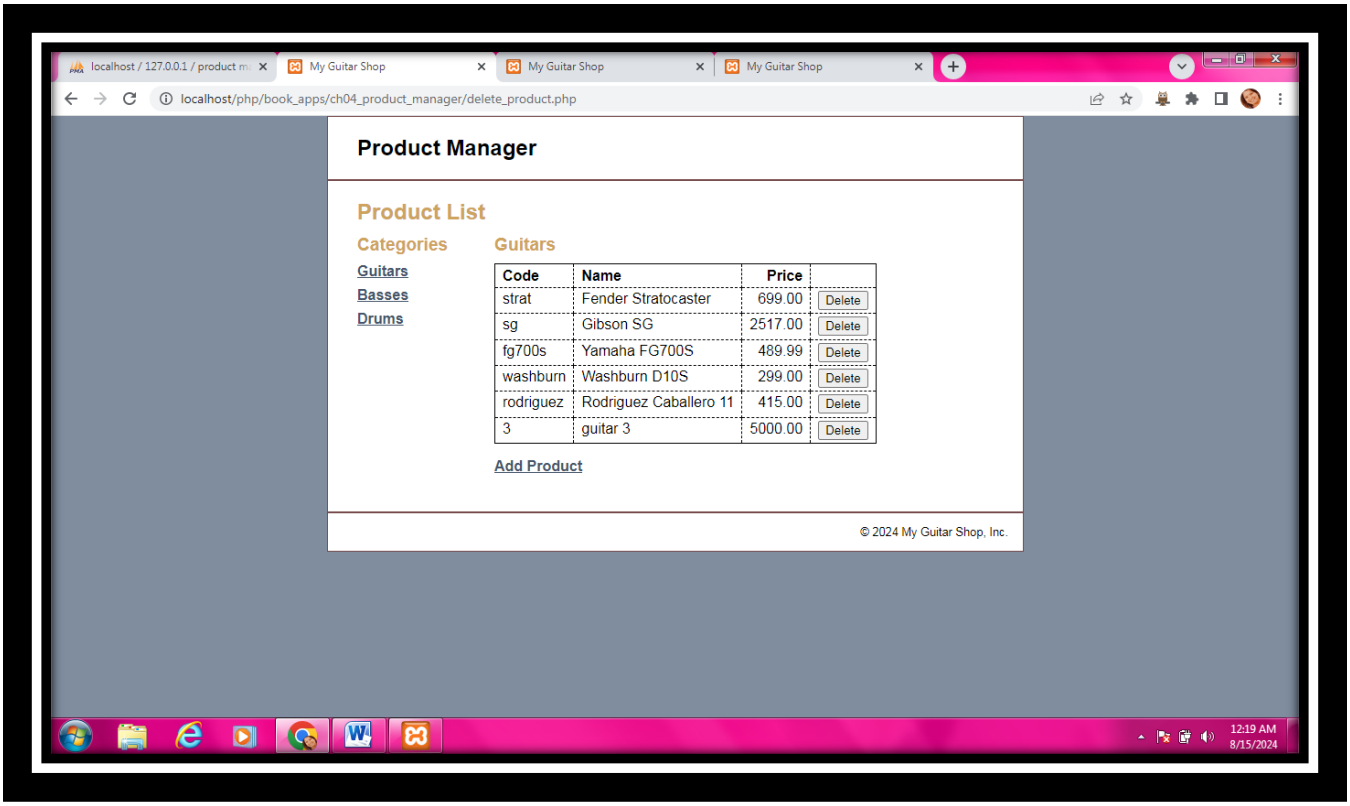



localhost / 127.0.0.1 / product m
My Guitar Shop
My Guitar Shop
My Guitar Shop
localhost/phpmyadmin/index.php?route=/database/import&db=product+manager
To get future Google Chrome updates, you'll need Windows 10 or later. This computer is using Windows 7.
Learn more
phpMyAdmin
Recent
Favorites
New
information_schema
my
mysql
performance_schema
phpmyadmin
product
product catalog
product manager
product viewer
test
Server: 127.0.0.1 » Database: product manager
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
Triggers
Importing into the database "product manager"
File to import:
File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in .[format].[compression]. Example: .sql.zip
Browse your computer: (Max: 40MiB)
Choose File
add_product.php
You may also drag and drop a file on any page.
Character set of the file:
utf-8
Partial import:
Console
Bookmarks
Options
History
Clear
Press Ctrl+Enter to execute query
12:15 AM
8/15/2024


localhost / 127.0.0.1 / product m
My Guitar Shop
My Guitar Shop
My Guitar Shop
localhost/phpmyadmin/index.php?route=/import
To get future Google Chrome updates, you'll need Windows 10 or later. This computer is using Windows 7.
Learn more
phpMyAdmin
Recent
Favorites
New
information_schema
my
mysql
performance_schema
phpmyadmin
product
product catalog
product manager
product viewer
test
Server: 127.0.0.1 » Database: product manager
Structure
SQL
Search
Query
Export
Import
Operations
Privileges
Routines
Events
More
Import has been successfully finished, 11 queries executed. (add_product.php)
Error
Static analysis:
6 errors were found during analysis.
1. Variable name was expected. (near "?" at position 1)
2. Unexpected character. (near "[" at position 53)
3. Unexpected character. (near "]" at position 67)
4. Unexpected beginning of statement. (near "?" at position 1)
5. Unexpected beginning of statement. (near "php" at position 2)
6. Unrecognized statement type. (near "Get" at position 10)
SQL query: Copy
<?php // Get the product data $category_id = $_POST['category_id'];
MySQL said:
#1064 - You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right syntax to use near '<?
Console
Bookmarks
Options
History
Clear
Press Ctrl+Enter to execute query
12:16 AM
8/15/2024


localhost / 127.0.0.1 / product m
My Guitar Shop
My Guitar Shop
My Guitar Shop
localhost/php/book_apps/ch04_product_manager/
Warning: Undefined array key "category_id" in C:\xampp\htdocs\php\book_apps\ch04_product_manager\index.php on line 6
Product Manager
Product List
Categories
Guitars
Basses
Drums
Guitars
Code
Name
Price
strat
Fender Stratocaster
699.00
Delete
sg
Gibson SG
2517.00
Delete
fg700s
Yamaha FG700S
489.99
Delete
washburn
Washburn D10S
299.00
Delete
rodriguez
Rodriguez Caballero 11
415.00
Delete
Add Product
© 2024 My Guitar Shop, Inc.
12:12 AM
8/15/2024


localhost / 127.0.0.1 / product m
My Guitar Shop
My Guitar Shop
My Guitar Shop
localhost/php/book_apps/ch04_product_manager/add_product_form.php
Product Manager
Add Product
Category:
Guitars
Code:
3
Name:
guitar 3
List Price:
5000
Add Product
View Product List
© 2024 My Guitar Shop, Inc.
12:18 AM
8/15/2024



RESULT:

The Program Will Be Executed, The Output Is Displayed

4. Enhance guitar shop application

AIM:

Enhance the Guitar Shop Application

ALGORITHAM:

Step 1: Set Up Your XAMPP Environment

1. **Install XAMPP**: Download and install XAMPP from the official site.
2. **Start XAMPP**: Launch the XAMPP Control Panel and start the **Apache** and **MySQL** services.

Step 2: Create the Database and Table

1. **Open phpMyAdmin**: In your browser, navigate to http://localhost/phpmyadmin.
2. **Create Database**:
   - Click on the **Databases** tab.
   - Enter product_catalog as the database name and click **Create**.
3. **Create Table**: Use the following SQL to create a products table:
4. **Import SQL File (Optional)**: If you have a SQL file with initial data, you can import it by:
   - Clicking on the **product_catalog** database.
   - Going to the **Import** tab.
   - Selecting the SQL file and clicking **Go**.

Step 3: Create the PHP Application

1. **Set Up Project Directory**:
   - Navigate to the htdocs folder in your XAMPP installation (usually found in C:\xampp\htdocs).
   - Create a new folder named book_Aapps
2. **Create PHP Files**: Create the following files in the book_apps directory.

Step 4: Test the Application

1. **Open the Application**: In your browser, navigate to http://localhost/book_apps/ch5 _guitar _shop/index.php.
2. **View Products**: You should see a list of products (if you have any in the database).

Step 5: Create Sample Data (Optional)
You can add sample data directly via phpMyAdmin:

1. Go to the **products** table.
2. Click on the **Insert** tab.
3. Fill in the fields (name, description, price) and click **Go**.

Step 6: Additional Features (Optional)You can enhance the application by adding features like:

- **Edit and Delete Functions**: Create separate pages for editing and deleting products.
- **Search Functionality**: Implement a search feature to filter products by name or description.

4. Enhance Guitar Shop Application

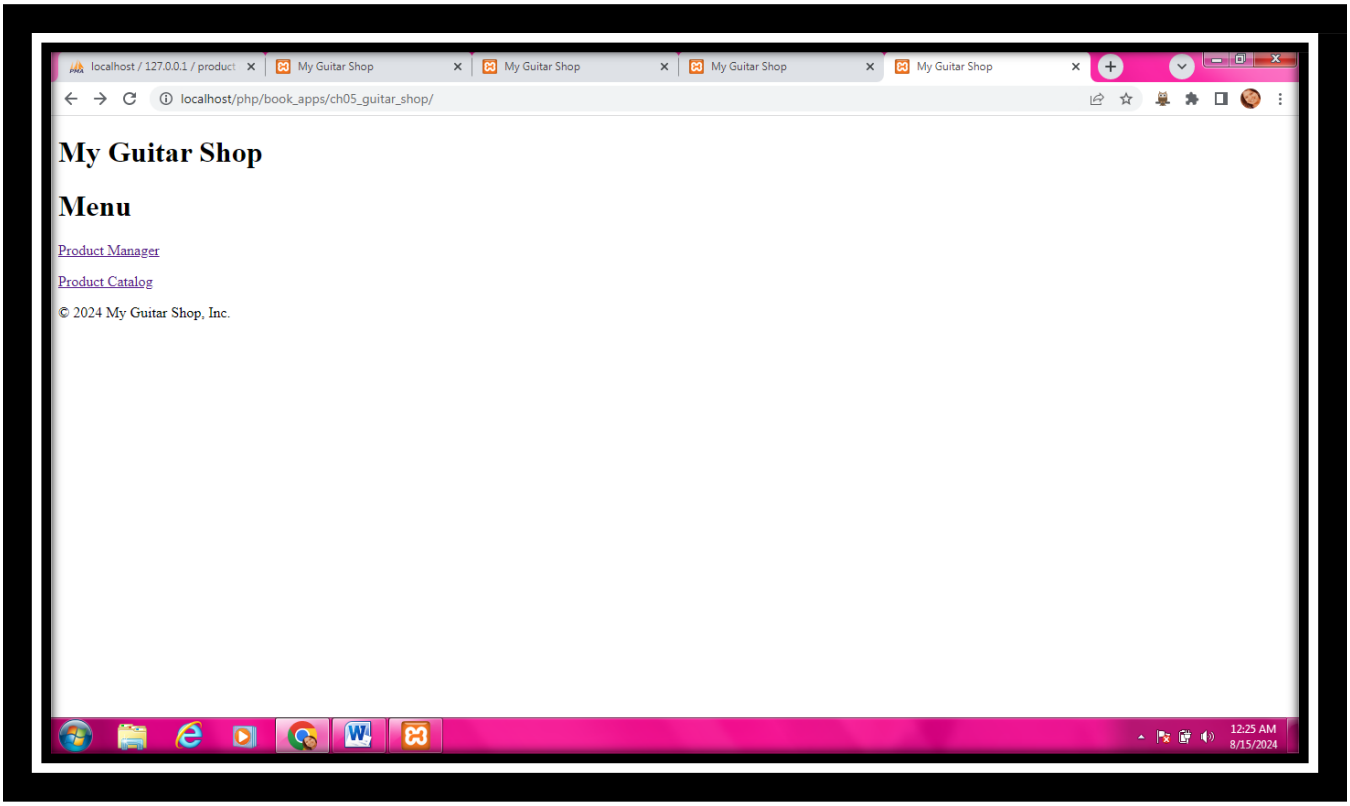
```
-- Create the database
        CREATE DATABASE my_guitar_shop1;


-- Select the database
        USE my_guitar_shop1;
-- Create categories table
        CREATE TABLE categories (
        categoryID INT AUTO_INCREMENT PRIMARY KEY,
        categoryName VARCHAR(255) NOT NULL
         );
-- Create products table
      CREATE TABLE products (
      productID INT AUTO_INCREMENT PRIMARY KEY,
      categoryID INT NOT NULL,
      productCode VARCHAR(10) NOT NULL,
      productName VARCHAR(255) NOT NULL,
      listPrice DECIMAL(10,2) NOT NULL,
      FOREIGN KEY (categoryID) REFERENCES categories(categoryID)
      ON DELETE CASCADE
      ON UPDATE CASCADE
);
```

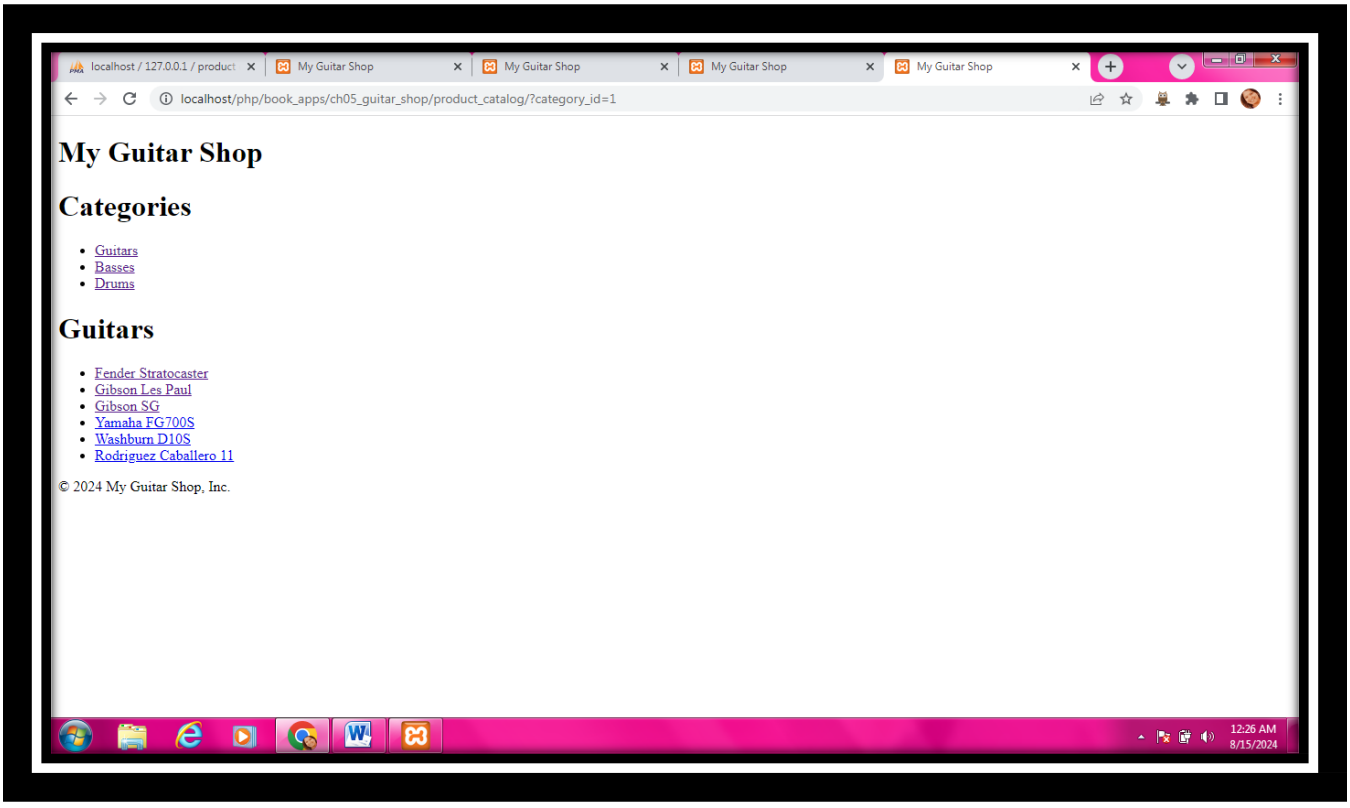
Database_connectivity

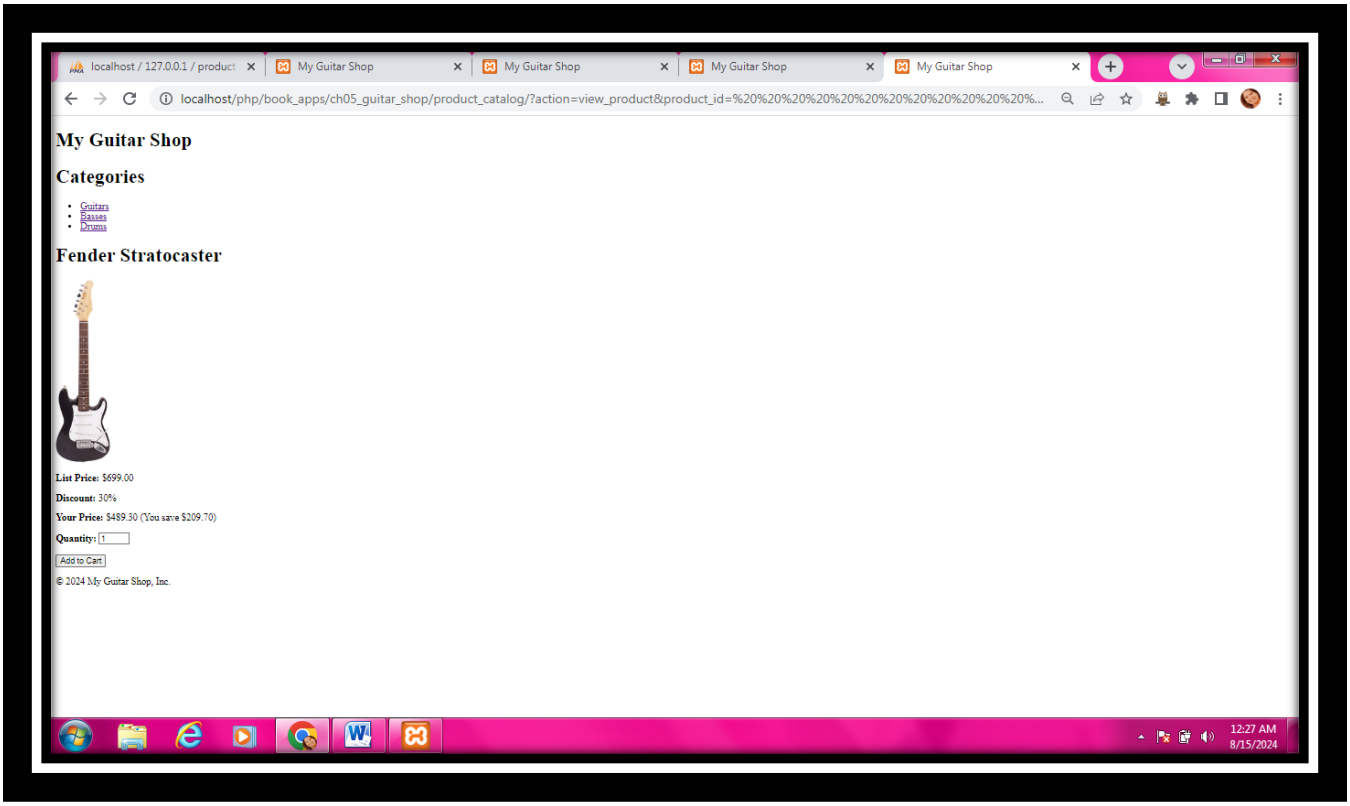
```
<?php
   $dsn = 'mysql:host=localhost;dbname=my_guitar_shop1';
   $username = 'root';
   $password = '';
   try {
      $db = new PDO($dsn, $username, $password);
   } catch (PDOException $e) {
      $error_message = $e->getMessage();
      include('database_error.php');
      exit();
   }
```

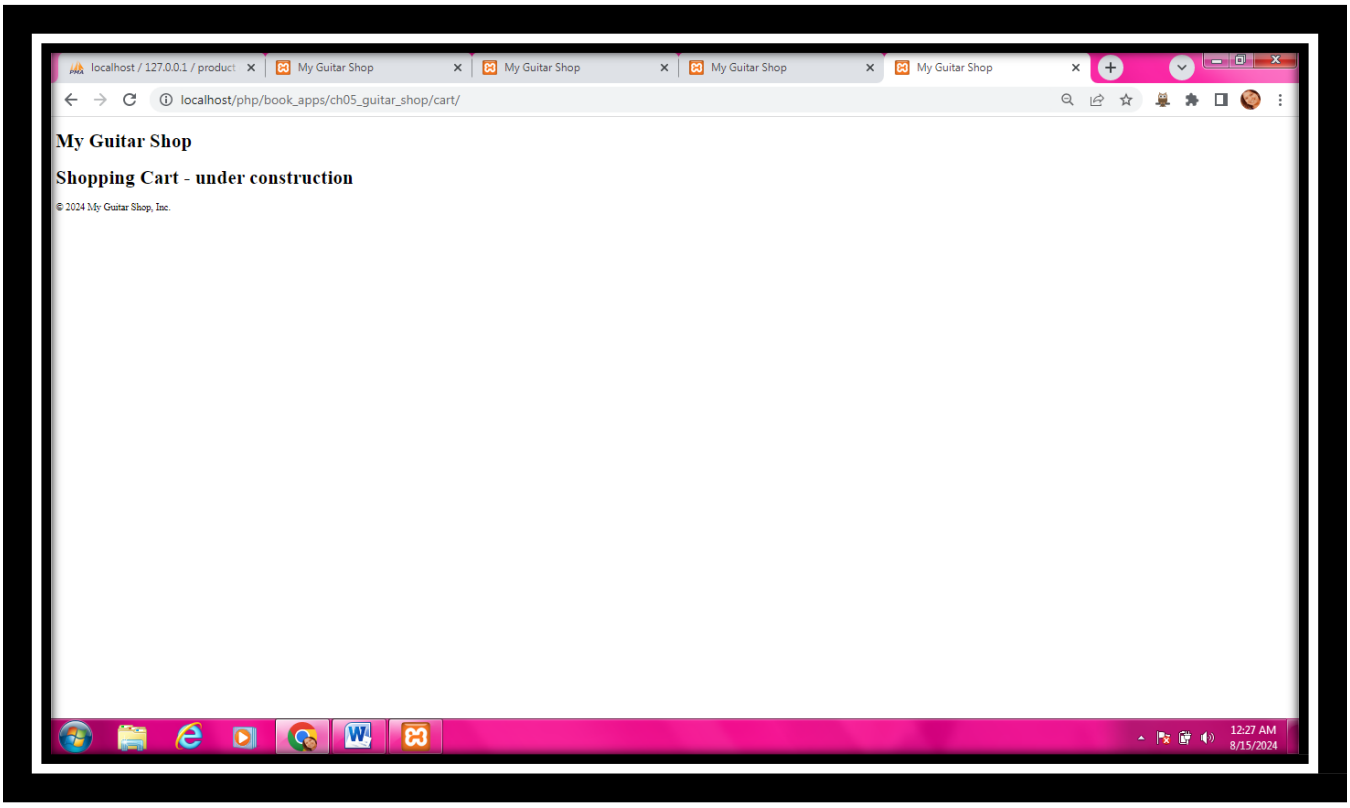
?>

OUTPUT:


localhost / 127.0.0.1 / product
My Guitar Shop
My Guitar Shop
My Guitar Shop
My Guitar Shop
localhost/php/book_apps/ch05_guitar_shop/
My Guitar Shop
Menu
Product Manager
Product Catalog
© 2024 My Guitar Shop, Inc.
12:25 AM
8/15/2024


localhost / 127.0.0.1 / product
My Guitar Shop
My Guitar Shop
My Guitar Shop
My Guitar Shop
localhost/php/book_apps/ch05_guitar_shop/product_catalog/?category_id=1
My Guitar Shop
Categories
Guitars
Basses
Drums
Guitars
Fender Stratocaster
Gibson Les Paul
Gibson SG
Yamaha FG700S
Washburn D10S
Rodriguez Caballero 11
© 2024 My Guitar Shop, Inc.
12:26 AM
8/15/2024





RESULT:

The Program Will Be Executed, The Output Is Displayed.

5. Fetching Category And Guitar Shop Database In PhpMyAdmin

AIM:

Write a select query to fetch category and product data of guitar shop database in phpMyAdmin

ALGORITHM:

Step1:categories.category_name: Fetches the name of the category.

Step2:products.product_name: Fetches the name of the product.

Step3:products.product_description: Fetches the description of the product.

Step4:products.price: Fetches the price of the product.

Step5:FROM products: Specifies that we are selecting data from the products table.

Step6:JOIN categories ON products.category_id = categories.id: Joins the products table with the categories table based on the foreign key relationship between category_id in the products table and id in the categories table.

Step7: This query will return a list of all products along with their corresponding category names. You can execute this query in the **SQL** tab of phpMyAdmin.

5. Fetching Category And Guitar Shop Database In PhpMyAdmin

To fetch category and product data from a guitar shop database in phpMyAdmin, you would typically have two tables: one for categories (categories) and one for products (products). These tables are usually linked by a foreign key, where the products table has a category_id that references the id in the categories table.

SQL SELECT query to fetch the category name and product details:

```
-- Products with category name

 SELECT p.productID, p.productCode, p.productName, p.listPrice, c.categoryName

FROM products p

JOIN categories c ON p.categoryID = c.categoryID;
```

OUTPUT:

Assuming the categories and products tables are populated with data, the result set might look something like this:

productID	productCode	productName	listPrice	categoryName
1	GTR001	Fender Stratocaster	799.99	Guitars
2	GTR002	Gibson Les Paul	1199.50	Guitars
3	KEY001	Yamaha Keyboard	1199.50	Keyboards
4	DRM001	Roland Drum Kit	699.25	Drums

RESULT:

The Program Will Be Executed, The Output Is Displayed.

6. Data Manipulation On The Product Table

AIM:

Perform Data Manipulation On The Product Table Of Guitar Shop Database In Php

MyAdmin

ALGORITHM:

Step1:Access phpMyAdmin

- Open your web browser and navigate to your phpMyAdmin URL. This is usually something like http://localhost/phpmyadmin if you're working locally.

Step2:Select the Database

- Once you're logged in, you will see a list of databases on the left-hand side. Click on the database that you want to manipulate data within.

Step3:Select the Table

- After selecting the database, you'll see a list of tables within that database. Click on the table where you want to manipulate data.

Step4:Browse Data

- To view existing data, click on the **"Browse"** tab. This will show you the records stored in the table.

Step5:Inserting Data

- To insert new data into the table, click on the **"Insert"** tab.
- Fill out the form fields corresponding to the columns in your table.
- Click **"Go"** to execute the insertion.

Step6:Editing Data

- In the **"Browse"** tab, find the row you want to edit and click on the **"Edit"** link next to it.
- Modify the data in the fields as needed.
- Click **"Go"** to save the changes.

Step7:Deleting Data

- In the **"Browse"** tab, find the row you want to delete and click on the **"Delete"** link next to it.
- phpMyAdmin will ask for confirmation before deleting the record. Confirm to proceed.

Step8:Executing SQL Queries

- You can manually run SQL queries by clicking on the **"SQL"** tab.
- Enter your SQL query (e.g., SELECT, UPDATE, DELETE, INSERT) in the provided text area.
- Click **"Go"** to execute the query.

Step9:Exporting Data

- To export data from a table, click on the **"Export"** tab.
- Choose the format (e.g., SQL, CSV) and other options as needed.
- Click **"Go"** to download the exported data.

6. Data Manipulation On The Product Table

In phpMyAdmin, data manipulation involves tasks such as inserting, updating, deleting, or selecting data from the database tables.

1. Inserting Data (INSERT)

Categories table

```
INSERT INTO categories (categoryName)
VALUES ('Guitars'), ('Keyboards'), ('Drums');
```

Products table

```
INSERT INTO products (categoryID, productCode, productName, listPrice)
VALUES  (1, 'GTR001', 'Fender Stratocaster', 799.99),
        (1, 'GTR002', 'Gibson Les Paul', 1199.50),
        (2, 'KEY001', 'Yamaha Keyboard', 499.00),
        (3, 'DRM001', 'Roland Drum Kit', 699.25);
```

2. Updating Data (UPDATE)

```
-- Change category name
UPDATE categories
SET categoryName = 'Electric Guitars'
WHERE categoryID = 1;

-- Update product price
UPDATE products
SET listPrice = 899.99
WHERE productCode = 'GTR001';

--Deleting Data (DELETE)
-- Delete one product
DELETE FROM products
WHERE productCode = 'DRM001';

-- Delete category (will also delete products under it because of ON DELETE CASCADE)
DELETE FROM categories
WHERE categoryID = 3;
```

4. Selecting Data (SELECT)

```
SELECT * FROM categories;

SELECT * FROM p
```

--

OUTPUT:

1. Inserting Data (INSERT)
1row inserted.

2. Updating Data (UPDATE)
1row affected.

3. Deleting Data (DELETE)
1row deleted.

4. Selecting Data (SELECT)

INSERT:CATEGORIES TABLE:

categoryID	categoryName
1	Guitars
2	Keyboards
3	Drums

INSERT:PRODUCTS TABLE:

ProductID	CategoryID	productCode	productName	listPrice
1	1	GTR001	Fender Stratocaster	799.99
2	1	GTR002	Gibson Les Paul	1199.50
3	2	KEY001	Yamaha Keyboard	499.00
4	3	DRM001	Roland Drum Ki	699.25

UPDATE|:CATEGORIES TABLE:

categoryID	categoryName
1	Electric Guitars
2	Keyboards
3	Drums

UPDATE|:PRODUCTS TABLE:

ProductID	productCode	productName	listPrice
1	GTR001	Fender stratocaster	899.99

DELETE: CATEGORIES TABLE:

categoryID	CategoryName
1	Electric Guitars
2	Key boards

DELETE : PRODUCTS TABLE

productID	prductCode	productName	listPrice
1	GTR001	Fender Stratocaster	899.99
2	GTR002	Gibson Les Paul	1199.50
3	KEY001	Yamaha Keyboard	499.00

RESULT:

The Program Will Be Executed, The Output Is Displayed.

7. Future Value Calculation Using A Function With Parameters

AIM:

Create a php program to perform future value calculation using a function with parameters

ALGORITHM:

Step1 : Start the APACHE & MYSQL server

Step2 : Open notepad

Step3 : Start the php script

Step4 : Define three variables

Step5 : The function future_value takes three parameters: PV (present value), r (interest rate), and n (number of periods).

Step6 : It uses the pow() function to raise (1 + $r) to the power of n.

Step7 : Finally, it returns the future value, which is displayed using number_format to format the output to two decimal places.

Step8 : Use the 'Echo' statement to view the output

Step9 : Save the program with .php extension in the Localhost/ xampp/htdocs/newfolder/p7.php

Step10: Go to chrome and run the program by localhost/foldername/p7.php

7. Future Value Calculation Using A Function With Parameters

```
<?php
function future_value($PV, $r, $n)
 {

// Calculate the future value

$FV = $PV * pow((1 + $r), $n);

return $FV;

}

// Example usage:

$PV = 1000;

// Initial investment of 1000

$r = 0.05;

// Annual interest rate of 5%

$n = 10;

// Investment period of 10 years


$FV = future_value($PV, $r, $n);

echo "The future value of the investment is: $" .number_format($FV, 2);

?>
```

OUTPUT:

The future value of the investment is: $1,628.89

RESULT:

The Program Will Be Executed, The Output Is Displayed.

8. GET_Array Method Using PHP

AIM:

Create PHP program for using GET Array

ALGORITHM:

Step 1: Start the APACHE & MYSQL server

Step 2 : Open the notepad

Step 3 : Open the php tag use of IF condition statement,

(isset($_GET[&fnmae,$lname,$submit]

Step 4 : Use $_GET method for getting the values

Step 5 : Use echo statement for printing

Step 6 : And declare else statement to specify when IF statement is not true

Step 7 : Close the PHP tag

Step 8 : Type the HTML program and use input tag for first name, last name Email and submit

Step 9 : Save the program write extension of php

Step 10: To run the program open browser LOCALHOST/FOLDER NAME

Step 11 : The output will be displayed

8. Get_Array Method Using Php

```
<?php
if(isset($_GET['submit']))
{
if($_GET["fname"] == "" || $_GET["lname"] == "" ){
echo "The firstname or lastname can't be empty";
}
else
{
echo "First Name: ". $_GET['fname']."<br/>";
echo "Last Name: ". $_GET['lname']."<br/>";
echo "Email: ". $_GET['email']."<br/>";
echo "Phone: ". $_GET['phone'];
}
}
else
{
?>
<html>
<head>
<title>Use of PHP $_GET</title>
</head>
<body>
<form method="get" action="#">
<table>
<tr><td>
<label for="inputName">Enter your first name:</label>
</td><td>
<input type="text" name="fname" id="fname"><br/>
</td></tr><tr><td>
<label for="inputName">Enter your last name:</label>
</td><td>
<input type="text" name="lname" id="lname"><br/>
</td></tr><tr><td>
<label for="inputName">Enter your email:</label>
</td><td>
<input type="text" name="email" id="email"><br/>
</td></tr><tr><td>
<label for="inputName">Enter your phone:</label>
</td><td>
<input type="text" name="phone" id="phone"><br/>
</td></tr><tr><td>
```

```
<input type="submit" name="submit" value="Submit"><br/>
</td><td></td></tr>
</table>
</form>
</body>
</html>
<?php
}
?>
```

Output:

First Name:santha
Last Name:sorubini
Email: santhasorubini@rvsgroup.com
Phone: 9790197054

RESULT:

The Program Will Be Executed, The Output Is Displayed.

9. POST_Array method using PHP

AIM:

Create the PHP program using POST Array method

ALGORITHM:

STEP-1 Start the APACHE & MYSQL server
STEP-2 Open the notepad
STEP-3 Open the php tag use of IF condition statement, (isset($_POST[&user name;$password;]
STEP-4 Use $_POST method for Passed the values
STEP-5 Use echo statement for printing
STEP-6 And declare else statement to specify when IF statement is not true
STEP-7 Close the PHP tag
STEP-8 Type the HTML program and use input tag for user name, password and submit
STEP-9 Save the program write extension of pho
STEP-10 To run the program open browser LOCALHOST/FOLDER NAME
STEP-11 The output will be displayed

9. POST_Array method using PHP

```
<?php
if(isset($_POST['submit']))
{
if(trim($_POST["username"]) == "admin" && trim($_POST["pass"]) == "238967" ){
echo "Authenticated User";
}
else
{
echo "Invalid user";
}
}
else
{
?>
<!-- HTML form to take input from the user -->
<html lang="en">
<head>
<title>Use of PHP $_POST</title>
</head>
<body>
<form method="post" action="#">
<table>
<tr><td>
<label for="inputName">Username:</label>
</td><td>
<input type="text" name="username" id="uname"><br/>
</td></tr><tr><td>
<label for="inputName">Password:</label>
</td><td>
<input type="password" name="pass" id="pass"><br/>
</td></tr><tr><td>
<input type="submit" name="submit" value="Submit"><br/>
</td><td></td></tr>
</table>
</form>
</body>
</html>
<?php
}
```

?>

OUTPUT:

Username: min

Password: ******

Submit

Authenticated User

OUTPUT :

Username: admin

Password: ****

Submit

Invalid user

RESULT:

The Program Will Be Executed, The Output Is Displayed.

10. Relational And Logical Operators

AIM:

Create PHP program for Relational and Logical operators

ALGORITHM:

.**Step1:** Start the APACHE & MYSQL server

Step2: Open notepad

Step3: Start the <?php script

Step4:DECLARED THE VALUES OF VARIABLE($X=70;$Y=30).

Step5: Used For If Condition ($X<=100&&$Y<=50) With Using Logical

Operator <= And Relational Operator&&

Step6: Use The 'Echo' Statement To View The Output .

Step7: Save The Program With .Php Extension In The Local Disk C >>Xampp>>Htdocs >> New Folder >>10.Php .

Step8: Go To Chrome And Run The Program By Localhost/Folder Name/10.php

10. Relational And Logical Operators

```
<html>
<head>Relational and logical operators</head>
<body>

<?php

// Example variables

$a = 10;
$b = 20;

// Relational and logical operators

if ($a < $b && $b > 15)
{
echo"<br>";
echo "a is less than b and b is greater than 15.";
}
elseif ($a == $b || $a > 5)
{
echo "a is equal to b or a is greater than 5.";
}
else
{
echo "None of the conditions were met.";
}
?>
</body>
</html>
```

OUTPUT:

Relational and logical operators

a is less than b and b is greater than 15.

RESULT:

The Program Will Be Executed, The Output Is Displayed.

11. Strings In PHP

AIM:

Create the PHP program for Strings

ALGORITHM:

Step1 : Start the APACHE & MYSQL server

Step2 : Open notepad

Step3 : Start the php script

Step4 : Define three variables

Step5 : $str1 contains a multi-line string enclosed in double quotes

Step6 : $str2 contains a multi-line string enclosed in double quotes with escaped double quotes

Step7 : $str3 contains a string with a escaped newline character('\n')

Step8: Use the 'Echo' statement to view the output of the 3 variables

Step9 : Save the program with .php extension in the Localhost/ xampp/htdocs/newfolder/p11.php

Step10 : Go to chrome and run the program by localhost/foldername/p11.php

11. Strings In PHP

```
<?php
$str1='Hello text multiple line text within single quoted string';
$str2='Using double "quote" directly inside single quoted string';
$str3='Using escape sequences \n in single quoted string';
echo "$str1 <br/> $str2 <br/> $str3";
?>
```

OUTPUT 1:

Hello text multiple line text within single quoted string
Using double "quote" directly inside single quoted string
Using escape sequences \n in single quoted string

Using ConcadinationAnother Code

```
echo "$str1 <br/> $str2"."$str3";
```

OUTPUT 2:

Hello text multiple line text within single quoted string
Using double "quote" directly inside single quoted string Using escape sequences \n in single quoted string

RESULT:

The Program Will Be Executed, The Output Is Displayed.

12. Numbers In PHP

AIM:

Create the PHP program for Numbers

ALGORITHM:

Step1 : Start the APACHE & MYSQL server

Step2 : Open notepad

Step3 : Start the <?php script

Step4 : Declare the variables $x

Step5 : Defined the different values in the variables

Step6 : Create the function var_dump

Step7 : Define the is_numeric format in the variables

Step8 : Close the PHP tag?>

Step9 : Save the program with .php extension in the Localhost/

xampp/htdocs/newfolder/p12.php

Step10: Go to chrome and run the program by localhost/foldername/p12.php

12. Numbers In PHP

```
<html>
<body>
<?php
// Check if the variable is numeric
$x = 5985;
var_dump(is_numeric($x));
echo "<br>";
$x = "5985";
var_dump(is_numeric($x));
echo "<br>";
$x = "59.85" + 100;
var_dump(is_numeric($x));
echo "<br>";
$x = "Hello";
var_dump(is_numeric($x));
?>
</body>
</html>
```

OUTPUT:

```
bool(true)
bool(true)
bool(true)
bool(false)
```

RESULT:

The Program Will Be Executed, The Output Is Displayed.

13. Built In Function

AIM:

Create PHP program for Built in Function

ALGORITHM:

Step 1 : Turn On The PC.

Step 2 : Open The **Xaamp**,And Start **APACHE**&**MYSQL**.

Step 3 : open **Notepad** to code the Function.

Step 4 : Start the code with **<?php**tag and and end with "**;**".

Step 5 : Use "**$**" Symbol to Name a Variable,Use**Sum** Function .

Step 6 : To Declare a Variable Using **$ Z = $X+$Y**.

Step 7 : Use **Echo** Statement To State An Values. **echo"x+y=".Sum(x,y);**.

Step 8 : Use **
**Tag , Give Values To a Statement .

Step 9 : Finish The Coding Function.

Step 10 : To Run a Program Open **Chrome** and Type **localhost/Folder Name**. And Select The File To View An Output In Parent Directory.

Step 11 : Stop The **APACHE**&**MYSQL** And Close The **XAMPP**.

13. Built In Function

```
<html>
<body>
<?php
function sum(int $x, int $y)
{
  $z = $x + $y;
  return $z;
}
echo "5 + 10 = " .sum(5,10) . "<br>";
echo "7 + 13 = " .sum(7,13) . "<br>";
echo "2 + 4 = " .sum(2,4);
?>
</body>
</html>
```

OUTPUT:

5 + 10 = 15
7 + 13 = 20
2 + 4 = 6

RESULT:

The Program Will Be Executed, The Output Is Displayed.

14.WHILE-LOOP IN PHP

AIM

Create PHP program for While Loop

ALGORITHM:

.**Step1:** Start the APACHE & MYSQL server

Step2: Open notepad

Step3: Start the <?php script

Step4:DECLARED THE VALUES OF VARIABLE [$x=0;] .

Step5: Used while loop to given the condition '($x<=100)' .

Step6: Use the 'echo' statement to view the output .

Step7: Save the program with .php extension in the local disk c >>xampp>>htdocs

>> new folder >>p14.php .

Step8: Go to chrome and run the program by localhost/folder name/p14.php

14.WHILE-LOOP IN PHP

```
<html>
<body>
<?php
$x = 0;
while($x <= 100)
{
  echo "The number is: $x <br>";
  $x+=10;
}
?>
</body>
</html>
```

OUTPUT:

The number is: 0
The number is: 10
The number is: 20
The number is: 30
The number is: 40
The number is: 50
The number is: 60
The number is: 70
The number is: 80
The number is: 90
The number is: 100

RESULT:

The Program Will Be Executed, The Output Is Displayed.

15. Array In PHP

AIM:

Create The PHP Program ForArray

ALGORITHM:

Step1: Start The APACHE &MYSQL Server

Step2: Open Notepad

Step3: StartThe <?Php Script

Step4: Declared The Array Values Of Variable.

Step5: Used For Loop To Given The Condition($Row=0;$Row<4;&Row++)

And Create The Second For Loop Condition(&Col=0;&Col<3;&Col++).

Step6: Use The 'Echo' Statement To View The Output .

Step7: Save The Program With .Php Extension In The Local Disk C >>Xampp>>Htdocs

>> New Folder >>p15.php .

Step8: Go To Chrome And Run The Program By Localhost/Folder Name/p15.php

15.Array In PHP

```
<html>
<body>
<?php
$cars = array (
  array("Volvo",22,18),
  array("BMW",15,13),
  array("Saab",5,2),
array("Land Rover",17,15)
);
for ($row = 0; $row < 4; $row++)
{
  echo "<p><b>Row number $row</b></p>";
  echo "<ul>";
  for ($col = 0; $col < 3; $col++)
{
    echo "<li>".$cars[$row][$col]."</li>";
  }
  echo "</ul>";
}
?>
</body>
</html>
```

OUTPUT:

Row number 0

- Volvo
- 22
- 18

Row number 1

- BMW
- 15
- 13

Row number 2

- Saab
- 5
- 2

Row number 3

- Land Rover
- 17
- 15

RESULT:

The Program Will Be Executed, The Output Is Displayed.